Cascade Non-Maximally Decimated Filter Banks Form Efficient Variable Bandwidth Filters for Wideband Digital Transceivers

*fred harrisa,Elettra Venosab, Xiaofei Chenc, Chris Dickd*

San Diego State University

San Diego, USA

afharris@mail.sdsu.edu, belettra.venosa@unina2.it, cchenxiaofei\_sdsu@yahoo.com, dchris.dick@xilinx.com

*Abstract*—This paper describes a number of techniques that use perfect reconstruction (PR) non-maximally decimated filter banks (NMDFBs) to implement variable bandwidth filters. These processes entail three tightly coupled tasks. The first task uses an M-path analysis filter bank to partition the full bandwidth input time series into a set of M reduced bandwidth, reduced sample rate, intermediate time series. The second task selects the subset of the M channel time series whose contiguous spectra spans the design bandwidth of the digital filter. The third task uses the M-path PR synthesis bank to assemble the desired output time series from the subset of selected channel time series. The first benefit of this process is that the computational workload of the cascade analysis-synthesis filter bank filter implementation is typically an order of magnitude below that of the tapped delay line, direct implementation, of the same filter. A second benefit is that a high data rate input time series is partitioned into a set of multiple, reduced sample rate, intermediate time series processed by reduced speed parallel arithmetic processors. This process enables simple, reduced cost, processing of input signals with GHz sample rates.

*Index Terms*—Non-maximally decimated filter bank, perfect reconstruction, polyphase channelizer, digital receiver, variable bandwidth filter.

1. INTRODUCTION

Many digital signal processing tasks can be performed with reduced computational burden when we embed a resampling operation in its implementation. N, the number of coefficients of a tapped delay line filter is determined by its specification parameters: specifically, sample rate (fSample), passband frequency (fPass) stopband frequency (fStop), passband ripple (δPass) and stopband ripple (δStop) as shown in (1a). A common approximation to this function is shown in (1b) where the transition bandwidth (Δf) is defined as the difference between stopband and passband frequencies.

 (1a)

 (1b)

The literature contains a number of approximations for the multiplier constant K(δPass, δStop) which are presented in equation form or more generally as graphical nomograms. Matlab’s *firpmord* function offers one approximation to the filter length from the filter parameter list. The important consideration here is that K increases when either or both ripple levels, δPass and δStop are decreased. N, the filter length is an important parameter because it is an indicator of the computational workload required to implement the filter. We can think of the units of N as *operations/output sample* where operations (or ops) includes, multiplies, adds, operand fetches, and address increments. We want small values of N. Once N is determined from (1b) there is no way to reduce N without relaxing the filter performance specifications, Δf, δPass, and δStop. There is, however, one more adjustable parameter in (1b), the sample rate fSample. If the filter reduces the output signal bandwidth, we can reduce the output sample rate in proportion to the bandwidth reduction, say by M, as shown in (2) where we also see that the units of N/M, is operations per input sample.

 (2)

The workload per input sample is smaller than the workload per output sample because the output workload is distributed over the M input samples. This reduced workload occurs without relaxing the performance specifications while still satisfying the Nyquist criterion for the output signal at the reduced sample rate.

Interestingly, we can reduce the sample rate at the input to the filter with an M-path filter partition. Doing so, we violate the Nyquist criterion with M-fold aliasing on the way into the filter and find at the output summing port we cancel the alias terms and have an aliased free output signal. Performing the workload to compute one output sample over M input samples offers significant computational savings which is often impressive. It may be that the application in which the filter is operating requires the output sample rate to be the same as the input sample rate. There are numerous cascade architectures we can propose that perform an input down sampling followed by an output up sampling. The one we illustrate is a cascade of 3 filters, an M-path input filter to reduce the sample rate M-to-1, an inner filter to perform the specified filtering task at the reduced sample rate, and an M-path output filter to increase the sample rate 1-to-M. In this architecture the input and output filters need not be formed from the original filter but can be designed to facilitate efficient resampling operations [1, 2, 4]. We illustrate this workload reduction by examining a filter with a large ratio of sample rate to bandwidth, one for which we can reduce sample rate as part of the processing. Table 1 presents such a set of specifications. Evaluating (1b) with the filter parameters of table 1 with K from a Hermann monogram, estimates the filter length at 800 taps as shown in (4). Matlab’s estimate was 771 taps. We increased the filter length slightly to 809 taps to accommodate a modified Remez algorithm’s 1/f decreasing stopband side-lobes

Table 1: Specification for FIR Filter

|  |  |
| --- | --- |
| Sample Rate | 2.0 MHz |
| Pass Band Frequency Span | 0-to-10 kHz |
| Stop Band Frequency Span | 20-to-1000 kHz |
| Pass Band Ripple | 0.05 dB (±5.8 10-3) |
| Stop Band Attenuation | -90 dB (3.16 10-5) |

 (4)



Figure 1. Spectra: 809 Tap Low Pass Filter and Cascade 120 Tap 20-to-1 Input FIR Filter, 50 tap Inner FIR Filter and 120 Tap 1-to-20 Output FIR Filter: and Passband Ripple Detail of Two Filters

The top subplot of Figure 1 shows the spectrum of the 809 tap direct implementation of the filter specified in Table 1 and the center subplot shows the spectrum of a cascade of three filters whose architecture is shown in figure 2. The three cascade is formed by a 120 tap filter partitioned into 20 path filter with 6 coefficients per path that processes 20 input samples to form 1 output sample. The inner filter operating at 100 kHz, 1/20-th of the input sample rate performs the actual spectral shaping matching the passband and stopband specifications of table 1. Since this inner filter is designed to operate at fs/20, its 40 tap length is about 1/20-th of the direct implementation designed to operate at fs. Not only is this filter shorter, it is operating at 1/20-th of the sample rate so the workload reduction is 1/400-th of the workload of the direct implementation. The third filter in the cascade is the dual of the input 120-tap filter, a 20 path filter with 6-coefficients per path that processes 1 input sample with 20 different path filters to output 20 output samples. The workload for the three filter cascade if 14 ops per input-output sample, a significant saving over the direct implementation of the 809 tap filter.



Figure 2. 120 Tap, 20-Path 20-to-1 Down-Sample Filter, 40-Tap Inner Filter and 120 Tap, 20-Path 1-to-20 Up-Sample Filter with Spectra at Input and Outputs if Inner and Output Filter.

# Synthesizing Wide bandwidth Filters

In the previous section we demonstrated that when a filter has a large ratio of sample rate to bandwidth the cascade of down sampling and up sampling filters offers substantial computational advantages over the direct implementation. It would appear that access to this computational advantage is not available for filters with bandwidth occupying a significant fraction of the sample rate. In fact, with a small change in perspective, we learn that similar efficiencies can be had for filters with wide bandwidths.

Our first step to access the wide bandwidth option is to partition the wide bandwidth spectrum into multiple narrow bandwidth channels [4-6]. Each narrowband channel will inherit the computational advantage of the single channel process. We accomplish the spectral partition into multiple narrow band channels by following the M-path down sampling filter with a bank of phase rotators to unwrap all the aliased spectral components from the integer multiples of the output sample rate as a result of the down sampling. The bank of phase rotators are equivalent to an inverse DFT or for their computational advantage an IFFT. The cascade of the M-path Filter and the M-point IFFT forms an M-channel analysis channelizer. To permit perfect reconstruction in the dual M-path up sampling channelizer the filter weights of the down sampling filter are designed to form Nyquist channels with frequency responses crossing their adjacent channel responses at their -6 dB levels. To obtain maximum flexibility of the cascade channelizers we double the output sample rate of the M path filters from fs/M to 2fs/M. This converts the filter bank from a maximally decimated to a non-maximally decimated filter bank (NMDFB) [3, 7, 8]. We do this to simplify the perfect reconstruction of the input spectrum from the outputs of the input M-path filter bank. The M/2-to-1 down sampling in the NMDFB causes the odd indexed Nyquist zones to alias to the half sample rate fs/M while the even indexed Nyquist zones alias to baseband. Circular shift of the input buffer to the IFFT reverses the phase of the odd indexed Nyquist zone aliases and places all aliased spectra at baseband.

Details of this minor modification can be found in a number of related papers.



Figure 3: Block Diagram of Channelizer Based Variable Bandwidth Filter. BW

Adjusted by Binary Mask between Analysis and Synthesis Filter Banks



Figure 4. Impulse Response of Super Channel Synthesized from 21 sub-Channels (top), Frequency Response of Synthesized Channel with Analysis Channelizer Channels Responses (center), and Zoom Detail of Passband Ripple (bottom)

Figure 3 shows the cascade of the analysis and synthesis filter banks with the bandwidth selection process performed by the binary mask between the two filter banks. Figure 4 illustrates by example how the masking operator between the cascade channelizers synthesizes a target filter with reduced workload. Here we see the spectrum of a targeted wide bandwidth filter implemented with 1300 taps. Rather than implement the wide band target filter directly, we implement a 1300 tap narrow band channelizer filter with a 6 dB bandwidth 1/100th of the sample rate but otherwise with the same transition bandwidth and same stopband attenuation. We then partition this filter into a 100-path filter which is coupled to a 100-point IFFT. The polyphase partition performs a 50-to-1 down sample and outputs 100 narrow band channels whose spectral centers and widths span the full Nyquist interval defined by the input sample rate. A subset of 21 of these channels is enabled to define the synthesized pass band frequencies and the remaining subset of these channels is disabled to define the synthesized filter stop band.

The output synthesis filter bank synthesizes the output time series from the selected channels enabled by the binary mask. The 1300 tap filter embedded in the 100 path filter contains 13 coefficients per path. Two paths are operated per input sample, so the input filter requires 26 operations per input sample. The 100 point IFFT following the 100 path filter is implemented by a hybrid mixed radix 4 by 25 Good-Thomas and Cooley-Tukey IFFT. When the 4, and 5 by 5 point transforms in the GT are implemented by Winograd short transform algorithms, the 100 point complex IFFT requires 550 real multiplies per complex output vector. These 550 multiplies are distributed over 50 input samples which adds 11.0 multiplies per input sample point for a total of 37 multiplies per input for input processing. The output process that performs the up-sampling and perfect reconstruction channelization also requires 37 multiplies per output sample. Thus the workload for the cascade filter bank is seen to be 74 operations per input output sample points. Comparing the computational workload for the 1300 tap filter implemented in the two modes, without and with resampling we find the resampled version of the filter requires 5.7% of the conventional implementation. In short the resampled version of the filter offers a green implementation of the filter.

# III. Improved Frequency Resolution-Option I

The cascade polyphase filter banks can synthesize broad bandwidth filters with from any even or odd integer multiple of the channel bandwidth. The example in the previous section formed a filter from 21 20-MHz channels. Now suppose we want additional flexibility in bandwidth selection. Say we want a filter with bandwidth spanned by a non-integer multiple of channel bandwidths such as 20.5 20-MHz channels instead of 21 20-MHz channels. We present a number of methods to accommodate the requirement for finer or arbitrary increments of synthesized bandwidths. In the first method, we alter the bandwidth of the two edge channels with an inner tier filter in a manner similar to that illustrated in figure 2. There we altered the single baseband channel’s bandwidth, but here we apply the same concept to the two edge filters of the selected channel span as shown in figure 4. The two inner tier filters have complex conjugate weights that form offset bandwidth reducing filters [10]



Figure 4. Very Efficient Two-Tier Channelizer Based High Resolution Variable Bandwidth Filter Synthesizer

.



Figure 5. Spectra of Channelizer Channel Responses, Inner Tier Filter Applied to End Channel, and Modified End Channel Response to Obtain Desired Synthesized Filter With Reduced Bandwidth.



Figure 6. Spectra of Synthesized Broadband Channel with Inner Tier Filters Reducing End Channel Bandwidth to Reduce Synthesized Bandwidth

between the output of the analysis filter and the input to the synthesis filter as shown in figure 5. These filter operate at the reduced sample rate, 2fs/M, and thus have a small number of coefficients with workload distributed over M/2 input samples. The workload to implement the inner filter is a small fraction of the channelizer work, typically increasing the workload of the cascade channelizers by only a few percent. The effect of the edge channel bandwidth reduction is shown in figure 6 for three examples. Note that each sample delay of the inner tier filter is responsible for M/2 sample delays at the output of the synthesizer channelizer output. We examine other options that reduce bandwidth without increasing system delay.

# Improved Frequency Resolution-Option-II

Rather than alter the bandwidth of only the edge filters, we can alter the bandwidth, but not center frequency, of all the filters in the analysis filter bank. We increase (or decrease) the bandwidth of the even indexed channel filters by an amount β while decreasing (or increasing) the bandwidth of the odd indexed channel filters by the same amount. The channelizer so formed will have two sets of interleaved complementary bandwidth channels as those shown in figure 7. The complementary channel bandwidths are formed in two analysis filter banks and then interleaved and binary masked in a single synthesis filter bank as shown in figure 8. The effect of the variable complimentary channel bandwidths is shown in figure 9 for three examples. Since we only require alternate bins from the two length M analysis filters in figure 8, we can pre-process the outputs of the two M-path filters to form butterfly outputs to implement a single M length analysis IFFT with interleaved channel filter outputs.

# Improved Frequency Resolution-Option-III

In the previous section we modified the bandwidth of the edge channel filters by modifying the bandwidth of all the channels filters. We accomplished this by forming two filter sets with wider and with narrower bandwidths and interleaved them with ordering that placed the narrow band channel at the band edge



Figure 7. Perfect Reconstruction Spectra of Adjacent Channel Widths of Modified Channelizer with Interleaved Complementary Bandwidth Channels



Figure 8. Synthesizing Interleaved Narrow and Wide Channel Channelizer from Dual Analysis Channelizers.



Figure 9. Spectra of Synthesized Broadband Channel with Interleaved Alternating Wide and Narrow Band Width Channel Filters

filter position. To change the band edge filter bandwidth we have to alter and redesign the impulse response of both prototype filters. In this final version of the variable band width edge filter options we form a hybrid option. We form one channelizer, the primary, with equal bandwidth channels which form all the channel filters except the band edge pair and design a second channelizer, the secondary, forms the pair of variable



Figure 10. Synthesizing Output Channelizer from Primary Major Channel Analysis Channelizer and Secondary Band Edge Channel Analysis Channelizers.



Figure 11. Spectra of Synthesized Broadband Channel with Secondary Band Edge Filters forming and Inserting End Channel Reduced Bandwidth Channels

band edge channel filters. The architecture of this third option is shown in figure 10. Here we see the primary (top) channelizer, processes the M output sample vector from the M-path fixed bandwidth filter with an M-point IFFT to form the major channel components of the synthesized output filter. We also see that the secondary (bottom) channelizer processes the M-output sample vector from the variable bandwidth filter with a single DFT vector to form the pair of band edge filter components of the variable bandwidth synthesized output filter. This filter option differs from the first option in that rather than reduce the bandwidth of the edge channel filter with a pair of inner low-pass filters, here we replace the band edge filters with alternate analysis filters formed in a secondary channelizer that operates with a DFT vector instead of the IFFT based phase de-spinners. The effect of the edge channel bandwidth reduction by this option is shown in figure11 for three examples. This form of the synthesized filter does not exhibit the additional output delay inserted by the first option

# Review and Concluding Comments

In this paper we have reviewed efficient architectures that use perfect reconstruction, non-maximally decimated polyphase analysis and synthesis filter banks to synthesis wide band filters with variable bandwidth with significant workload reduction relative to the direct tapped delay line implementation of the same filter. The process synthesizes filter bandwidths that are integer multiples of the analysis channel bandwidth. We then reviewed two options and introduced a new option for modifying the bandwidth of the synthesized filter by altering the channel bandwidth of the band edge filters [11, 12]. One option inserted an inner, bandwidth reducing, filter in the signal transfer path of the band edge channels. This filter inserted delay in the edge channel path which required matching delay in the remaining enabled channel paths. A second option interleaved PR filters with alternating wider and narrower channel bandwidths to span the selected channel span. The bandwidth of the synthesized filer is widened or narrowed by the channel filter at the band edge location. The third option alters the bandwidth of the only the band edge filter by synthesizing it in secondary variable bandwidth channelizer operating in parallel with the primary channelizer rather than in series as did the first option. This option did not insert additional processing delay and only requires changing one filter to alter the synthesized wide band filter

# References

[1] f.j.harris; C. Dick; X. Chen, E. Venosa, "Wideband 160-channel polyphase filter bank cable TV Channelizer", *Signal Processing, IET*, vol. 5, no. 3, pp. 325,-332, June 2011.

[2] f.j. harris, E. Venosa, X. Chen and B.D. Rao, “Polyphase Analysis Filter Bank Down-Converts Unequal Channel Bandwidths with Arbitrary Center Frequencies,” *Analog Integr Circ Sig Process*, vol. 71, no. 3, pp. 481-494, 2012.

[3] T. Karp and N. J. Fliege, "Modified DFT filter banks with perfect reconstruction", *IEEE Trans. Circuits Syst. II, Anal. Digital Signal Process.,*vol. 46, no. 11, pp. 1404 -1414, Dec. 1999.

[4] Chen, X.; harris, f.j.; Venosa, E.; Rao, B.D., "Non-Maximally Decimated Analysis/Synthesis Filter Banks: Applications in Wideband Digital Filtering,"*Signal Processing, IEEE Transactions on*, vol. 62, no. 4, pp. 852, 867, Feb. 15, 2014.

[5] f.j. harris, “*Multirate Signal Processing for Communication Systems*”, Prentice Hall, Upper Saddle River, New Jersey 07458, 2004.

[6] Vaidyanathan, P.P., "Theory and design of M-channel maximally decimated quadrature mirror filters with arbitrary M, having the perfect-reconstruction property," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 35, no. 4, pp. 476, 492, Apr 1987.

[7] Vetterli, M.; Le Gall, D., "Perfect reconstruction FIR filter banks: some properties and factorizations," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 37, no. 7, pp. 1057, 1071, Jul 1989.

[8] Jain, V.K.; Crochiere, R.E., "A novel approach to the design of analysis/Synthesis filter banks", *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '83.*, vol. 8, no. 3, pp. 228, 231, Apr 1983.

[10] f. harris, C. Dick, X. Chen, and E. Venosa, “M-Path Channelizer with Arbitrary Center Frequency Assignments”, WPMC 2010, Oct 11, 2010 - Oct 14, 2010, Recife, Brazil.

[11] f. harris, B. Behrokh, X. Chen, and E. Venosa, “Multi-resolution PR NMDFBs for Programmable Variable Bandwidth Filter in Wideband Digital Transceivers”, Int’l Conf. on DSP-2014, Hong Kong, 20-23 August 2014

[12] f. harris, C. Dick, X. Chen, and E. Venosa, “Interleaving Different bandwidth Narrowband Channels in Perfect reconstruction Cascade polyphase Filter banks for Efficient Flexible Variable bandwidth Filters in Wideband Digital Transceivers”, DSP-2015 Conf., Singapore, July 2015.